**1. Overloading the main Method**

In Java, the main method can be overloaded just like any other method. Overloading allows you to have multiple main methods with different parameter lists.

**Example:**

java

Copy code

public class MainOverload {

public static void main(String[] args) {

System.out.println("Original main method");

main(10); // Calls the overloaded main method

}

public static void main(int arg) {

System.out.println("Overloaded main method with int: " + arg);

}

public static void main(String arg) {

System.out.println("Overloaded main method with String: " + arg);

}

}

**Output:**

sql

Copy code

Original main method

Overloaded main method with int: 10

**Notes:**

* The JVM always looks for the public static void main(String[] args) method as the program entry point.
* Other main methods are treated as normal static methods and won't be invoked automatically by the JVM.

**2. Overriding the main Method**

You **cannot override** the main method in the true sense because:

1. The main method is static, meaning it belongs to the class, not an instance. Static methods cannot be overridden.
2. You can define a main method in a subclass, but it hides the superclass's main method instead of overriding it.

**Example:**

java

Copy code

class SuperClass {

public static void main(String[] args) {

System.out.println("Main method in SuperClass");

}

}

class SubClass extends SuperClass {

public static void main(String[] args) {

System.out.println("Main method in SubClass");

}

}

**Running:**

* If you run SuperClass, the output is:

css

Copy code

Main method in SuperClass

* If you run SubClass, the output is:

css

Copy code

Main method in SubClass

**Explanation:**

* The main method in SubClass does not override SuperClass's main. Instead, it hides it, as both methods are static.

**Key Points**

| **Feature** | **Overloading main** | **Overriding main** |
| --- | --- | --- |
| **Possible?** | Yes | Not in the true sense (only hiding). |
| **Use Case** | Useful for testing or additional logic | Rarely used, typically avoided. |
| **Entry Point** | JVM calls main(String[] args) | JVM doesn't care about subclasses. |